Dobývání znalostí z databází

8. října 2024

Základní pojmy

Data mining ([dejta majnyn], angl. dolování z dat či vytěžování dat) je analytická metodologie získávání netriviálních skrytých a potenciálně užitečných informací z dat. Někdy se chápe jako analytická součást dobývání znalostí z databází (knowledge discovery in databases, KDD), jindy se tato dvě označení chápou jako souznačná. Často dochází také k překryvu s termínem data science, který bývá obvykle chápán v širší míře než data mining.

Data mining se používá v komerční sféře (například v marketingu při rozhodování, které klienty oslovit dopisem s nabídkou produktu), ve vědeckém výzkumu (například při analýze genetické informace) a v jiných oblastech (například při monitorování aktivit na internetu s cílem odhalit činnost potenciálních škůdců a teroristů).

Metodologie data miningu

Protože data mining zahrnuje velkou šířku metod a způsobů práce, je obtížné podat jednoznačný návod k postupu. Přesto během 90. let vykrystalizovaly dvě obecné metodologie, které alespoň v hrubých rysech popisují jednotlivé kroky: metodologie *SEMMA*, za níž stojí firma SAS, a *CRISP-DM*, vyvinutá konsorciem firem, mezi něž patřil druhý hlavní hráč na trhu, SPSS.

Společnou podstatou všech metodologií je následnost několika kroků:

- **Obchodní/praktický** – formulace úlohy a porozumění problému. Ani automatické vyhledávání znalostí nelze provádět zcela naslepo.
- **Datový** – vyhledání a příprava dat pro analýzu. Statistické algoritmy většinou potřebují data připravená v určité podobě, a proto není možné použít přímo surových dat z obchodních databází.

2. Dobývání znalostí z databází

- **Analytický** – hledání informace v datech, vytváření statistických modelů a podobně. Využívají se nejrůznější metody od jednoduchých tabelací a vizualizací až po sofistikované přístupy jako je genetické programování nebo simulované žíhání. Asi nejčastěji používanými metodami však jsou logistická regrese s automatickým výběrem proměnných, rozhodovací stromy a neuronové sítě. Výstup této fáze bývá dvojí: Jednak obecnější *znalosti* (např. že svobodní klienti nejčastěji nakupují pozdě večer, zatímco ženatí po obědě), jednak matematické *modely* (např. postup, jak vytipovat potenciálního klienta pro daný produkt, např. termínovaný vklad).
- **Aplikační** – zjištěné poznatky a modely je třeba uvést do praxe, například spuštěním reklamní kampaně nebo reorganizací webových stránek.
- **Kontrolní** – je třeba zajistit zpětnou vazbu (jak efektivní byla obchodní akce) a v případě dlouhodobě nasazovaných modelů i kontrolovat, zda model příliš nezestárl a zachovává si svoji efektivitu.

Potenciální nebezpečí data miningu

Protože komerční data mining představuje často masivní a inteligentní zpracování osobních údajů, vznikají často obavy ze zneužití těchto informací.

Kromě obvyklých negativ spojených se shromažďováním osobních údajů, jako je záměrný i nezáměrný únik dat a jejich využití k různým nečestným aktivitám od spamu až po vydírání, teoreticky hrozí i různá zneužití statistických technik.

Zdá se však, že toto nebezpečí je – alespoň v současném stavu data miningu – nepatrné. I kdyby se náhodou zločinci dostali k využitelným osobním datům, pravděpodobně by jim použití sofistikovaných statistických metod příliš nepomohlo, už proto, že by jim chyběla databáze „pozitivních příkladů“ úspěšných zločinů, na níž by mohli své modely postavit.

Za větší potenciální nebezpečí lze považovat technologie, k jejichž vzniku data mining přispívá v akademické sféře. Například dekódování genomu může být použito k nehumánním selekcím osob podobným eugenice, ale postaveným na vědeckém základě. Anebo pokročilé metody identifikace osob mohou být spolu s kamerovými systémy používány ke špehování pohybu občanů.

Související pojmy

Machine learning (strojové učení) – část procesu modelování zpracování dat, která se zabývá technikami a algoritmy umožňujícími systému „učit se“.

Data science – termín obdobný data miningu, není zcela přesně ukotven, nahrazuje některé starší pojmy (business analytics).

Umělá inteligence – schopnost strojů vykazovat inteligentní chování; v současné době buzzword, tento termín je (neprávem) spojován především s celou řadou aplikací hlubokých neuronových sítí.

Business intelligence – proces analyzování a reportování historických dat.

Používané techniky data miningu

- rozhodovací stromy
- asociační pravidla
- neuronové sítě
- regresní analýza
- shluková analýza (pro marketingovou segmentaci)

Problémy

Ze správných dat, použijeme-li správný způsob dobývání, dostaneme správné výsledky. Proto by dobývání dat mělo být založeno na správných datech. Nicméně protože se používají statistické metody, tak výsledky jsou tzv. statisticky správné, tj. s velmi malou pravděpodobností odvodíme chybné výsledky (falešně pozitivní).

Dobývání typicky neodhalí všechny souvislosti schované v datech.

I z nesmyslných dat, například špatně připravených anebo náhodných dat, dostaneme nějaké výsledky. Je možné dostat i výsledky, co vypadají smysluplně, ale jsou smetí dovnitř, smetí ven (angl. Garbage In, Garbage Out – GIGO).

Pojem „dolování dat“

Data Mining (neboli dolování dat) je proces hledání informací a znalostí ve velkém objemu dat. Jedná se o nástroj používaný v rámci Business Intelligence (oblast analýzy dat sloužící jako podklady pro manažerské rozhodování). Pro data mining se využívají technologie rozpoznávání vzorů, statistické a matematické metody.

Definice Data Miningu

Data Mining je proces výběru, prohledávání a modelování ve velkých objemech dat, sloužící k odhalení dříve neznámých vztahů mezi daty za účelem získání obchodní výhody. (Fayyad 2010)

Jinak: Dolování dat je hledáním skrytých souvislostí, procesem výběru, prohledávání a modelování ve velkých objemech dat. Slouží k odhalení dříve neznámých vztahů mezi daty.

Metody dolování dat

Dnes užívanými metodami dolování dat jsou například:

- regresní metody (lineární regresní analýza, nelineární regresní analýza, neuronové sítě)
- klasifikace (diskriminační analýza, logistická regresní analýza, rozhodovací stromy, neuronové sítě),
- segmentace – shlukování (shluková analýza, genetické algoritmy, neuronové shlukování – Kohonenovy mapy)
- analýza vztahů (asociační algoritmus pro odvozování pravidel typu „if X then Y“)
- predikce v časových řadách (Boxova-Jenkinsonova metoda, neuronové sítě, autoregresní modely, ARIMA)
- detekce odchylek

Modely dolování dat

Deskriptivní model – popisuje nalezené vzory a vztahy v datech, které mohou ovlivnit rozhodování, např. analýza prodeje zboží v supermarketu, na jejímž základě je pak umístěno zboží v regálech.

Prediktivní model – umožňuje předvídat budoucí hodnoty atributů na základě nalezených vzorů v datech, např. analýza zákazníků, u kterých je vysoká pravděpodobnost, že budou reagovat na písemnou reklamní nabídku apod.

Co je overfitting (přeučení) modelu ?

Čím je způsobeno a jak mu zabránit ?

- Přeučení modelu u data miningu
- Naučený model je příliš svázán s trénovacími daty
- Přesnost modelu je vysoká na trénovacích datech, ale nízká na nových datech

Jak zabránit přeučení modelu ?

1. Rozdělení trénovacích dat (učení – test)
2. Rozhodovací stromy – prořezávání, menší hloubka stromu
    1. Některé algoritmy ukončí včas generování stromu (prepruning)
    2. Většina nejdříve vygeneruje strom a pak ho ořeže (postpruning)
    3. Prořezávání zvyšuje chybu na učící množině, ale doufáme, že na reálných datech chybu zmenší

Dobývání znalostí (především z databází)

Dobývání znalostí (KDD, Knowledge Discovery in Databases). Tento pojem definujeme jako proces netriviálního objevování implicitních, dopředu neznámých a potenciálně použitelných znalostí v datech. Podle této definice se někteří jedinci mohou mylně domnívat, že KDD je synonymum pro „data mining“, ovšem není tomu tak.

Data mining je pouze jedním krokem z celkového procesu dobývání znalostí z databází, a to proces vyhledávání požadovaných dat a jejich poskytování uživateli/procesu k dalšímu využití.

Definice KDD

Knowledge Discovery In Databases (KDD) je proces netriviálního objevování implicitních, dopředu neznámých a potenciálně použitelných znalostí v datech. (Fayyad 2010)

Oba pojmy (data mining a KDD) tedy víceméně znamenají totéž, u KDD je považována jako důležitá i samotná příprava dat.

Dolování dat je úzce spojeno s pojmem datový sklad. Pro dolování je důležitá kvalita vstupních dat; datové sklady obvykle obsahují data, která jsou už v určité míře předzpracovaná a očištěná od chyb.

Při analýze pomocí dolování dat často není dopředu známo, zda budou získány použitelné výsledky.

Rovněž interpretace získaných výsledků je považována za nejnáročnější fázi dobývání znalostí.

Procesy KDD

Na rozdíl od prostého využití statistických metod a metod strojového učení se v procesu KDD klade důraz na přípravu dat pro analýzu a interpretaci. Nejprve byl vytvořen tzv. technologický pohled na proces dobývání znalostí a poté díky Ananda a kol. byl v roce 1996 vytvořen tzv. manažerský proces dobývání znalostí, který se nyní používá ve většině případů.

Technologický proces KDD

1. **Selekce**: Příprava z dat, uložených v datovém skladu či tabulkách informačního systému, vede k vytvoření jedné tabulky, která obsahuje relevantní údaje o sledovaných objektech.

2. **Předzpracování**: Dochází k odstranění odlehlých hodnot, vyčištění datových chyb a výpočtu odvozených hodnot. Tato část procesu dobývání znalostí z databází může být vysoce výpočetně náročná.
3. **Transformace**: Dochází k diskretizaci spojitých veličin, například pro metody pracující s entropií jednotlivých atributů.
4. **Data mining**: Zde se používají vybrané analytické metody pro nalezení souvislostí v datech. Jedná se o výpočetně a paměťově nejnáročnější část celého procesu. Během procesu se využívají metody různých druhů v závislosti na předchozích výsledcích analytických metod.
5. **Interpretace**: Zpracováváno je velké množství výsledků jednotlivých metod. Některé výsledky lze přímo interpretovat, jiné se musí upravit do podoby srozumitelné pro uživatele.

Manažerský proces KDD

Vše začíná reálným problémem. Cílem procesu KDD je získat co nejvíce relevantních řešení daného problému.

1. **Řešitelský tým**: Expert na řešenou tématiku, expert na data a expert na metody dobývání znalostí z databází.
2. **Specifikace problému**: Máme určitý problém, který se týká určité skupiny lidí s určitými potřebami. Musíme specifikovat určité položky či si rozdělit zákazníky do různých skupin A a B.
3. **Získání dat**: Posouzení všech dostupných dat, zda jsou relevantní k danému problému. Nemusí se pracovat jen s nejnovějšími daty, ale také s daty déle archivovanými.
4. **Výběr metod**: Existuje několik typů metod dolování dat. Při každé analýze jsou potřebné jiné metody. Pokaždé se pracuje

s jiným množstvím metod, které se kombinují či nahrazují jinými. Mezi používané typy metod patří: klasifikační metody, klasické metody explorační analýzy dat, rozhodovací stromy, genetické algoritmy, Bayesovské sítě, neuronové sítě atd...

5. **Předzpracování dat**: Příprava dat do formy vyžadované pro aplikaci vybraných metod.
6. **Data mining**: Aplikace vybraných metod pro hledání vztahů mezi daty. Vždy závisí na jednotlivých předchozích výsledcích a podle nich se aplikují různé metody.
7. **Interpretace**: Aplikováním různých metod získáme značné množství výsledků. Některé výsledky mohou být již známé či nepodstatné, jiné se dají okamžitě aplikovat. Jednotlivé výsledky jsou často uspořádány do analytické zprávy (není to však podmínkou).

Úlohy KDD

1. Klasifikace/predikce: Cílem je nalézt znalosti potřebné pro klasifikaci nových případů. Získané znalosti musejí co nejvíce odpovídat danému konceptu. U predikce hraje důležitou roli čas (ze starších hodnost se odhaduje budoucí vývoj – např. předpověď počasí či pohyb akcií).
2. Deskripce: Cílem je najít dominantní strukturu nebo vazby, které jsou skryty v datech. Jsou požadovány pouze srozumitelné znalosti. Je dána přednost menšímu množství méně přesných znalostí.
3. Hledání nuggetů: Požadujeme nové zajímavé znalosti, které nemusí pokrýt daný koncept.

Užití KDD

Velmi často se KDD využívá v bankovnictví. Pracuje se s různými kritérii, jako je pohlaví, zaměstnanost, příjem, pravidelnost plateb atd. KDD banky využívají také pro zvyšování bezpečnosti platebních a kreditních karet, kdy jsou monitorovány činnosti uživatelů a každé "vybočení z normálu" banka ověřuje. Další využití KDD je v boji proti "praní špinavých peněz".

Ideální oblastí je ovšem internetové prostředí, které má obrovskou datovou kapacitu. Na internetu běží mnoho projektů, které se snaží zvýšit kvalitu ve vyhledávání (Web Mining, Web Content Mining a Web Usage Mining). Velkou roli hraje KDD také při přeměně internetu na sémantický web.

Software pro KDD

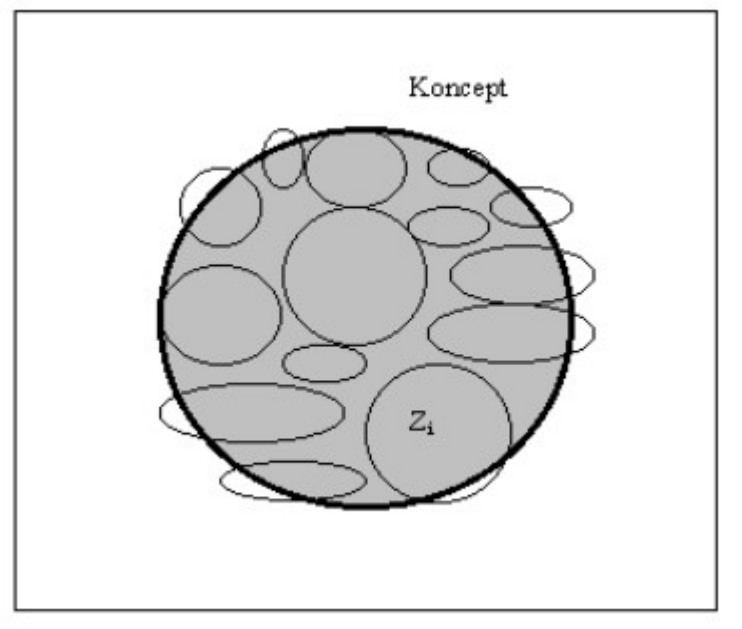
V literatuře se označuje **Decission Support System** (DSS). Nejznámějšími softwarovými produkty v této oblasti jsou Enterprise Miner (od firmy SAS) a Clementine (firma SPSS). Existují i nekomerční produkty jako Weka, Orange, Tanagra atd.

V České republice se systémy DSS zabývá například firma Adastra nebo projekt Lisp-Miner (VŠE Praha).

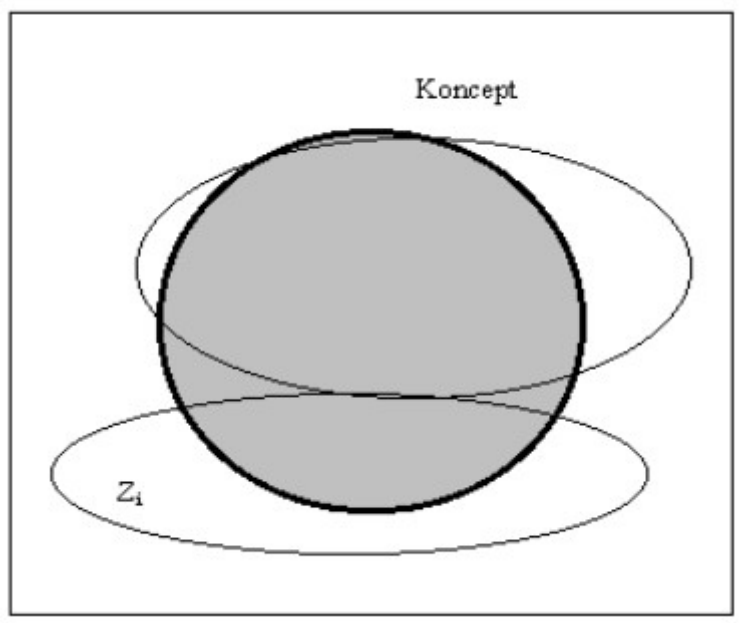
Úlohy dobývání znalostí

Hrubší členění (Klosgen, Zytkow, 1997):

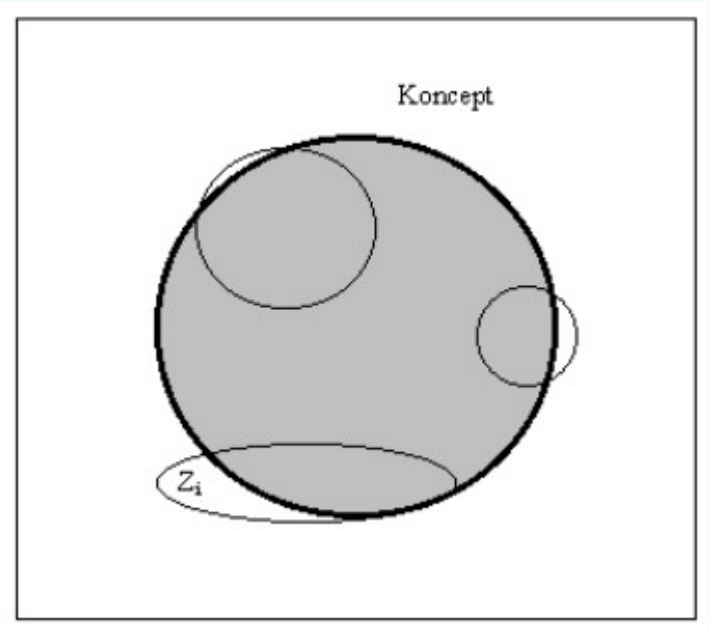
- klasifikace / predikce: cílem je nalézt znalosti použitelné pro klasifikaci nových případů



- deskripce: cílem je nalézt dominantní strukturu nebo vazby



- hledání „nuggetů“: cílem je nalézt dílčí překvapivé znalosti



Jemnější členění úloh dobývání znalostí (Chapman a kol., 2000):

- deskripce dat a jejich sumarizace
- segmentace dat
- deskripce konceptů
- klasifikace konceptů
- predikce vývoje dat a znalostí
- analýza závislostí v datech, příp. ve znalostech
- detekce odchylek v datech a znalostech

Metodologie dolování dat

Cílem metodologií je poskytnout uživatelům jednotný rámec pro řešení různých úloh z oblasti dobývání znalostí. Tyto metodologie umožňují sdílet a přenášet zkušenosti z úspěšných projektů.

Nejpoužívanější metodologie jsou: SEMMA (SAS), 5A (SPSS) a CRISP-DM.

CRISP-DM

CRISP (CRoss Industry Standard Proces for Data Mining) je souhrnná metodologie dobývání znalostí z databází, umožňuje provádět rozsáhlé projekty dobývání rychleji, efektivněji a méně nákladně prostřednictvím osvědčených postupů.

Základní etapy procesu dobývání dat jsou:

- Co řešit (Business understanding) – porozumění problematice, formulování úlohy
- Kde vzít data (Data understanding)

- Jak data připravit (Data preparation)
- Jak data analyzovat (Data modelling)
- Co jsme zjistili (Evaluation) – porozumění výsledkům
- Jak výsledky využít (Deployment)

SEMMA

Podle metodologie SEMMA spočívá proces dobývání v těchto krocích:

- Sample – vybírání vhodných objektů
- Explore – vizuální explorace a redukce dat
- Modify – seskupování objektů a hodnot atributů, datové transformace
- Model – analýza dat
- Assess – porovnání modelů a interpretace

5A

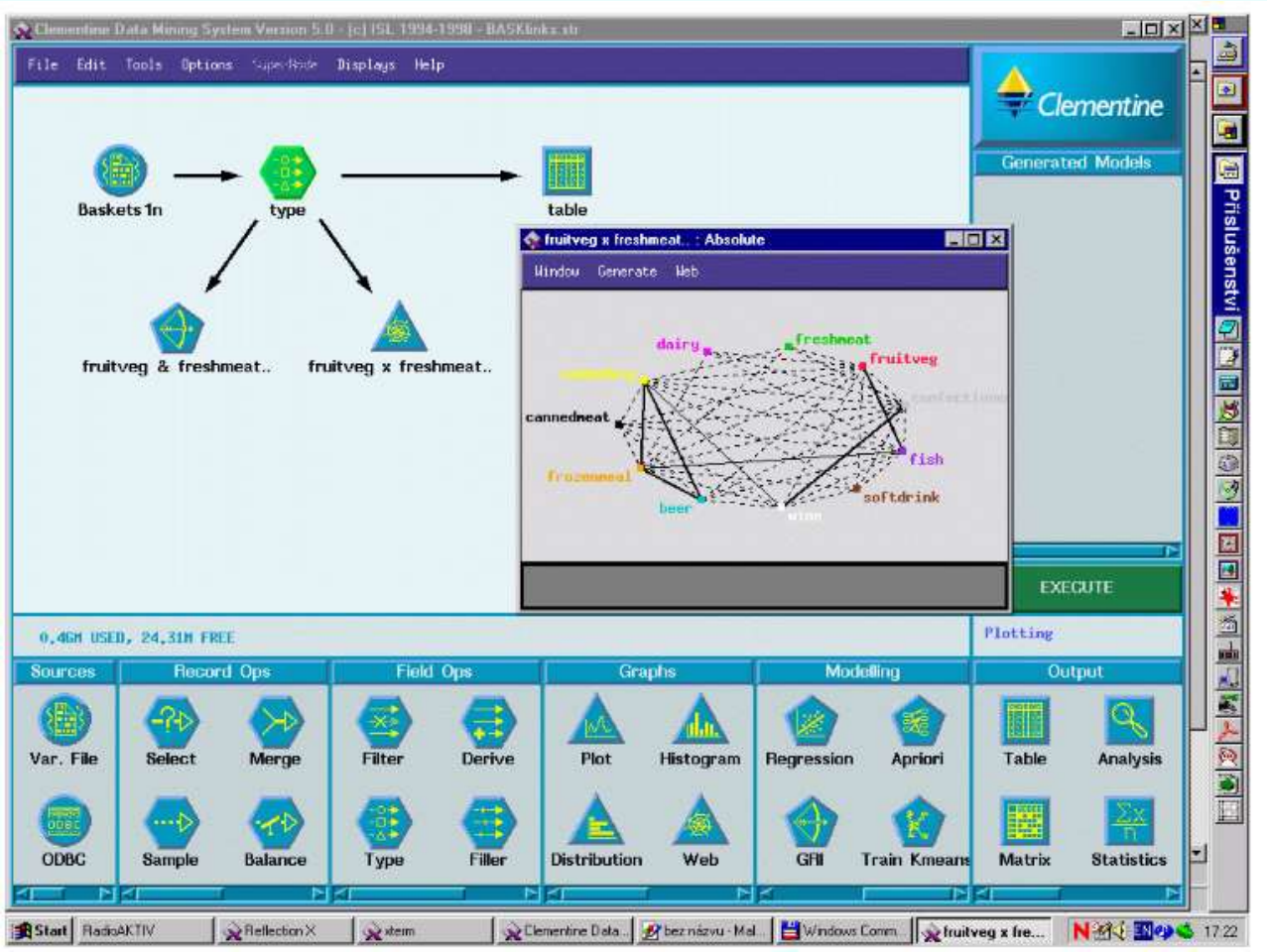
- Assess – posouzení potřeb projektu
- Access – shromáždění potřebných dat
- Analyze – provedení analýz
- Act – přeměna znalostí na akční znalosti
- Automate – převedení výsledků analýzy do praxe

Aplikační oblasti dobývání znalostí

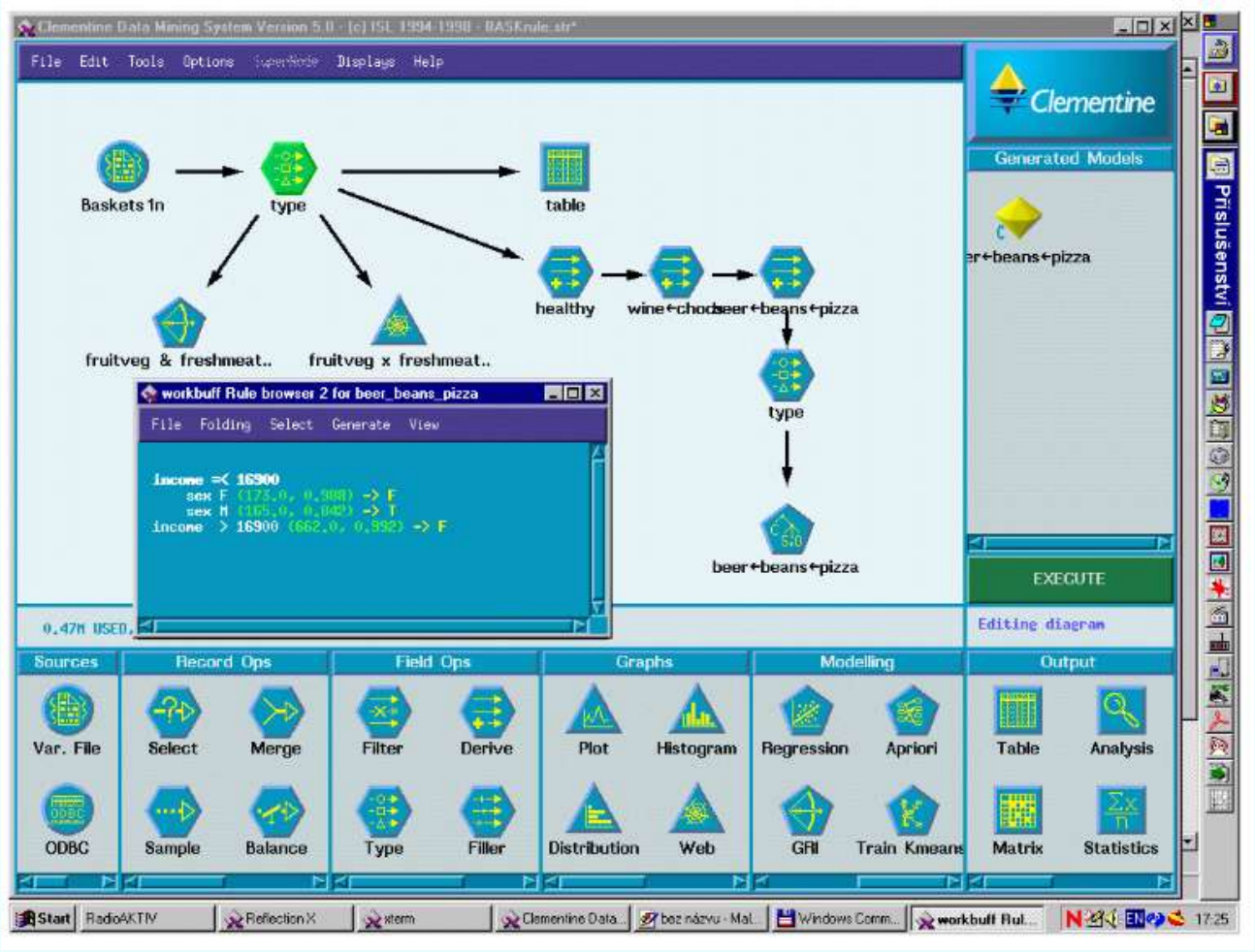
- segmentace a klasifikace klientů banky (např. rozpoznání problémových nebo naopak vysoce bonitních klientů)
- predikce vývoje kursů akcií
- predikce spotřeby elektrické energie
- analýza příčin poruch v telekomunikačních sítích
- analýza důvodů změny poskytovatele nějakých služeb (internet, mobilní telefony)
- segmentace a klasifikace klientů pojišťovny
- určení příčin poruch automobilů
- rozbor databáze pacientů v nemocnici
- analýza nákupního košíku (Market Basket Analysis)

Př.: Analýza nákupního košíku: pohled na data

Př.: Analýza nákupního košíku: deskripce



Př.: Analýza nákupního košíku: klasifikace



Standardy pro zápis modelů dolování dat

Predictive Modeling Markup Language

Standard na bázi XML vyvinutý v Data Mining Group (www.dmg.org), který slouží pro popis dat, datových transformací a vytvořených modelů dat.

Základní části PMML dokumentu:

- Header
- Data Dictionary
- Data Transformations
- Model

PMML: A markup language for data mining

Last Updated: 2022-11-28

Predictive Model Markup Language (PMML) is a markup language for data mining. Based on XML, PMML provides a standard that enables data mining models to be shared between the applications of different vendors. It provides a vendor-independent method of defining models. With PMML, you can exchange models between different applications.

For more information about PMML, visit Data Mining Group.

- **Intelligent Miner conformance with PMML**
  Intelligent Miner® supports PMML up to version 3.2. The term PMML refers to PMML V3.2 and previous versions of PMML unless otherwise specified.
- **PMML consumer conformance clause**
  Intelligent Miner provides an SQL interface that enables the application of PMML models to data.
- **Sources of PMML models**
  Intelligent Miner applies PMML models to data tables in Db2 databases. In the Db2 databases, the PMML models are stored as binary large objects (BLOBs).

Predictive Model Markup Language (PMML) is an XML-based language that provides a standardized way to represent and share predictive models between different applications, systems, and platforms. Developed by the Data Mining Group (DMG), PMML aims to facilitate the interoperability and deployment of predictive models, making it easier for data scientists, statisticians, and developers to collaborate and integrate their work into various applications and processes.

PMML supports a wide range of predictive modeling techniques, including but not limited to:

1. Regression models: Linear regression, logistic regression, and generalized linear models.
2. Decision trees: Classification and regression trees, as well as ensemble methods like random forests and gradient boosting machines.
3. Neural networks: Multilayer perceptron and radial basis function networks.
4. Clustering models: k-means, hierarchical clustering, and DBSCAN.
5. Association rules: Apriori and Eclat algorithms for discovering frequent itemsets and association rules.
6. Text mining models: Text classification and sentiment analysis techniques.
7. Time series models: ARIMA, Exponential Smoothing State Space Model (ETS), and GARCH.

The main advantages of using PMML include:

1. Standardization: PMML provides a common language and format for representing predictive models, making it easier to share and exchange models between different tools and platforms.
2. Faster deployment: With PMML, data scientists can develop models using their preferred tools and programming languages, and then easily export those models in a standardized format. This can reduce the time and effort required to deploy models into production systems, as developers can directly import and use the PMML models without needing to re-implement or translate them.
3. Interoperability: PMML enables the integration of predictive models with various data processing and analytics platforms, such as databases, data warehouses, and big data ecosystems (e.g., Hadoop and Spark).
4. Reduced development effort: By leveraging PMML, developers can focus on integrating predictive models into applications and processes, rather than spending time and resources on model implementation and translation.

Despite its advantages, there are some limitations to using PMML:

1. Limited support for some advanced models: PMML may not fully support all the features and capabilities of some advanced modeling techniques, like deep learning and custom algorithms.
2. Evolving standard: As new modeling techniques and technologies emerge, the PMML standard needs to be updated and extended, which can lead to compatibility issues between different versions of the standard.

In summary, Predictive Model Markup Language (PMML) is an XML-based language that provides a standardized way to represent and share predictive models between different applications, systems, and platforms. PMML facilitates the interoperability and deployment of predictive models, making it easier for data scientists, statisticians, and developers to collaborate and integrate their work into various applications and processes. While PMML offers several advantages, there are also some limitations to consider when using the standard.

Example PMML Regression model

```
<PMML version="4.2" xsi:schemaLocation="http://www.dmg.org/PMML-4_2 http://www.dmg.org/v4-2-1/pmml-4-2.xsd"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://www.dmg.org/PMML-4_2">
  <Header copyright="JBoss"/>
  <DataDictionary numberOfFields="5">
    <DataField dataType="double" name="fld1" optype="continuous"/>
    <DataField dataType="double" name="fld2" optype="continuous"/>
    <DataField dataType="string" name="fld3" optype="categorical">
      <Value value="x"/>
      <Value value="y"/>
    </DataField>
    <DataField dataType="double" name="fld4" optype="continuous"/>
    <DataField dataType="double" name="fld5" optype="continuous"/>
  </DataDictionary>
  <RegressionModel algorithmName="linearRegression" functionName="regression" modelName="LinReg"
normalizationMethod="logit" targetFieldName="fld4">
    <MiningSchema>
      <MiningField name="fld1"/>
      <MiningField name="fld2"/>
      <MiningField name="fld3"/>
      <MiningField name="fld4" usageType="predicted"/>
      <MiningField name="fld5" usageType="target"/>
    </MiningSchema>
    <RegressionTable intercept="0.5">
      <NumericPredictor coefficient="5" exponent="2" name="fld1"/>
      <NumericPredictor coefficient="2" exponent="1" name="fld2"/>
      <CategoricalPredictor coefficient="-3" name="fld3" value="x"/>
      <CategoricalPredictor coefficient="3" name="fld3" value="y"/>
      <PredictorTerm coefficient="0.4">
        <FieldRef field="fld1"/>
        <FieldRef field="fld2"/>
      </PredictorTerm>
    </RegressionTable>
  </RegressionModel>
</PMML>
```

Example PMML Scorecard model

```
<PMML version="4.2" xsi:schemaLocation="http://www.dmg.org/PMML-4_2 http://www.dmg.org/v4-2-
1/pmml-4-2.xsd" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://www.dmg.org/PMML-4_2">
  <Header copyright="JBoss"/>
  <DataDictionary numberOfFields="4">
    <DataField name="param1" optype="continuous" dataType="double"/>
    <DataField name="param2" optype="continuous" dataType="double"/>
    <DataField name="overallScore" optype="continuous" dataType="double" />
    <DataField name="finalscore" optype="continuous" dataType="double" />
  </DataDictionary>
  <Scorecard modelName="ScorecardCompoundPredicate" useReasonCodes="true" isScorable="true"
functionName="regression"    baselineScore="15" initialScore="0.8"
reasonCodeAlgorithm="pointsAbove">
    <MiningSchema>
      <MiningField name="param1" usageType="active" invalidValueTreatment="asMissing">
      </MiningField>
      <MiningField name="param2" usageType="active" invalidValueTreatment="asMissing">
      </MiningField>
      <MiningField name="overallScore" usageType="target"/>
      <MiningField name="finalscore" usageType="predicted"/>
    </MiningSchema>
    <Characteristics>
      <Characteristic name="ch1" baselineScore="50" reasonCode="reasonCh1">
        <Attribute partialScore="20">
          <SimplePredicate field="param1" operator="lessThan" value="20"/>
        </Attribute>
        <Attribute partialScore="100">
          <CompoundPredicate booleanOperator="and">
```

```
            <SimplePredicate field="param1" operator="greaterOrEqual" value="20"/>
            <SimplePredicate field="param2" operator="lessOrEqual" value="25"/>
          </CompoundPredicate>
        </Attribute>
        <Attribute partialScore="200">
          <CompoundPredicate booleanOperator="and">
            <SimplePredicate field="param1" operator="greaterOrEqual" value="20"/>
            <SimplePredicate field="param2" operator="greaterThan" value="25"/>
          </CompoundPredicate>
        </Attribute>
      </Characteristic>
      <Characteristic name="ch2" reasonCode="reasonCh2">
        <Attribute partialScore="10">
          <CompoundPredicate booleanOperator="or">
            <SimplePredicate field="param2" operator="lessOrEqual" value="-5"/>
            <SimplePredicate field="param2" operator="greaterOrEqual" value="50"/>
          </CompoundPredicate>
        </Attribute>
        <Attribute partialScore="20">
          <CompoundPredicate booleanOperator="and">
            <SimplePredicate field="param2" operator="greaterThan" value="-5"/>
            <SimplePredicate field="param2" operator="lessThan" value="50"/>
          </CompoundPredicate>
        </Attribute>
      </Characteristic>
    </Characteristics>
  </Scorecard>
</PMML>
```

Example PMML Tree model

```
<PMML version="4.2" xsi:schemaLocation="http://www.dmg.org/PMML-4_2 http://www.dmg.org/v4-
2-1/pmml-4-2.xsd" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://www.dmg.org/PMML-4_2">
  <Header copyright="JBOSS"/>
  <DataDictionary numberOfFields="5">
    <DataField dataType="double" name="fld1" optype="continuous"/>
    <DataField dataType="double" name="fld2" optype="continuous"/>
    <DataField dataType="string" name="fld3" optype="categorical">
      <Value value="true"/>
      <Value value="false"/>
    </DataField>
    <DataField dataType="string" name="fld4" optype="categorical">
      <Value value="optA"/>
      <Value value="optB"/>
      <Value value="optC"/>
    </DataField>
    <DataField dataType="string" name="fld5" optype="categorical">
      <Value value="tgtX"/>
      <Value value="tgtY"/>
      <Value value="tgtZ"/>
    </DataField>
  </DataDictionary>
  <TreeModel functionName="classification" modelName="TreeTest">
    <MiningSchema>
      <MiningField name="fld1"/>
      <MiningField name="fld2"/>
```

```
    <MiningField name="fld3"/>
    <MiningField name="fld4"/>
    <MiningField name="fld5" usageType="predicted"/>
  </MiningSchema>
  <Node score="tgtX">
    <True/>
    <Node score="tgtX">
      <SimplePredicate field="fld4" operator="equal" value="optA"/>
      <Node score="tgtX">
        <CompoundPredicate booleanOperator="surrogate">
          <SimplePredicate field="fld1" operator="lessThan" value="30.0"/>
          <SimplePredicate field="fld2" operator="greaterThan" value="20.0"/>
        </CompoundPredicate>
        <Node score="tgtX">
          <SimplePredicate field="fld2" operator="lessThan" value="40.0"/>
        </Node>
        <Node score="tgtZ">
          <SimplePredicate field="fld2" operator="greaterOrEqual" value="10.0"/>
        </Node>
      </Node>
      <Node score="tgtZ">
        <CompoundPredicate booleanOperator="or">
          <SimplePredicate field="fld1" operator="greaterOrEqual" value="60.0"/>
          <SimplePredicate field="fld1" operator="lessOrEqual" value="70.0"/>
        </CompoundPredicate>
        <Node score="tgtZ">
          <SimpleSetPredicate booleanOperator="isNotIn" field="fld4">
            <Array type="string">optA optB</Array>
```

```
            </SimpleSetPredicate>
          </Node>
        </Node>
      </Node>
      <Node score="tgtY">
        <CompoundPredicate booleanOperator="or">
          <SimplePredicate field="fld4" operator="equal" value="optA"/>
          <SimplePredicate field="fld4" operator="equal" value="optC"/>
        </CompoundPredicate>
        <Node score="tgtY">
          <CompoundPredicate booleanOperator="and">
            <SimplePredicate field="fld1" operator="greaterThan" value="10.0"/>
            <SimplePredicate field="fld1" operator="lessThan" value="50.0"/>
            <SimplePredicate field="fld4" operator="equal" value="optA"/>
           <SimplePredicate field="fld2" operator="lessThan" value="100.0"/>
            <SimplePredicate field="fld3" operator="equal" value="false"/>
          </CompoundPredicate>
        </Node>
        <Node score="tgtZ">
          <CompoundPredicate booleanOperator="and">
            <SimplePredicate field="fld4" operator="equal" value="optC"/>
            <SimplePredicate field="fld2" operator="lessThan" value="30.0"/>
          </CompoundPredicate>
        </Node>
      </Node>
    </Node>
  </TreeModel>
</PMML>
```

Př.: Systémy pro dobývání znalostí z databází a meziroční nárůst používání:

Tool	% change	2018 % share	2017 % share
Keras	108%	22.2%	10.7%
PyTorch	92%	6.4%	3.4%
Amazon Machine Learning	74%	3.3%	1.9%
RapidMiner	65%	52.7%	31.9%
Other free analytics/data mining tools	53%	8.3%	5.4%
DeepLearning4J	39%	3.4%	2.4%
Anaconda	37%	33.4%	24.3%
PyCharm	33%	13.5%	10.1%
Tensorflow	32%	29.9%	22.7%
Excel	24%	39.1%	31.5%
Tableau	21%	26.4%	21.8%

Zdroj: https://www.kdnuggets.com/2018/05/poll-tools-analytics-data-science-machine-learning-results.html

Rozvoj systémů pro dobývání znalostí z databází

Mezi současnými hlavními trendy rozvoje systémů je možné identifikovat:

1. Automatizaci celého procesu, zpřístupnění méně technickým uživatelům
   - OptiML firmy BigML
   - RapidMiner, TurboProp, Auto Model
   - Weka: Auto-Weka
   - Kompletní automatizace procesu – Datarobot (AI Cloud Platform)
2. Rozšiřování cloudových platforem největších IT hráčů
   - Azure MachineLearning Studio – Data Science and ML Platform
   - Google Cloud Platform
   - Machine Learning on AWS (Amazon Web Services)

Literatura

RAGEL, Roshan: Difference between KDD and Data mining. In: Difference between.com. [online]. 1. 5. 2011. [cit. 2014-05-12]. Dostupné z: http://www.differencebetween.com/difference-between-kdd-and-vs-data-mining/

BERKA, Petr: Dobývání znalostí z databází. Vyd. 1. Praha: Academia, 2003, s. 15.17. ISBN 80-200-1062-9

MRÁZOVÁ, Iveta: Dobývání znalostí. Přednáška [online]. Praha, UK-MFF, [cit. 2014-05-13]. Dostupné z: http://ksvi.mff.cuni.cz/~mraz/datamining/lecture/Dobyvani_Znalosti_Prednaska_Uvod.pdf

LIŠKA, Miroslav. Dobývání znalostí z databází. Přednáška [online]. Ostrava, OU, 2008, s. 10 [cit. 2014-05-13]. Dostupné z: http://www1.osu.cz/~klimesc/public/files/DOZNA/Dobyvani_znalosti_z_databazi.pdf

A mnoho dalších publikací …